IN THE **CLAIMS:**

1.   (Currently Amended) A method of managing a an in-kernel HTTP cache in a web server, comprising:

receiving a HTTP request;

determining whether response data associated with the HTTP request is in the in-kernel HTTP cache;

when it is determined that response data associated with the HTTP request is in the in-kernel HTTP cache, obtaining an advisory state associated with the HTTP request from the in-kernel HTTP cache, the advisory state when in a first state indicating that it is necessary to obtain permission from a HTTP daemon to determine whether the response data can be transmitted and when in a second state indicating that the response data can be transmitted without obtaining permission from the HTTP daemon; and

transmitting the response data in accordance with the advisory state associated with the HTTP request.

2.   (Previously Amended) The method as recited in claim 1, wherein transmitting the response data in accordance with the advisory state associated with the HTTP request comprises:

when the advisory state is in the first state, sending an advisory request to the HTTP daemon, the advisory request requesting an advise state from the HTTP daemon indicating an action to be taken with the response data, and receiving the advise state from the HTTP daemon.

3.   (Currently Amended) The method as recited in claim 2, wherein transmitting the response data in accordance with the advisory state associated with the HTTP request when the advisory state is in the first state further comprises:

transmitting the response data without modifying the response data in the in-kernel HTTP cache when the advise state is in a first state.

4.   (Currently Amended) The method as recited in claim 2, wherein transmitting the response data in accordance with the advisory state associated with the HTTP request when the advisory state is in the first state further comprises:

modifying the response data stored in the in-kernel HTTP cache as specified by the advise state.

5.   (Currently Amended) The method as recited in claim 4, wherein modifying the response data comprises:

removing at least one of the response data and the advisory state from the in-kernel HTTP cache when the advise state is in a second state.

6.   (Currently Amended) The method as recited in claim 4, wherein modifying the response data comprises:

receiving second response data from the HTTP daemon; and

performing at least one of replacing the response data in the in-kernel HTTP cache with the second response data and replacing the advisory state in the in-kernel HTTP cache with a second advisory state when the advise state is in a third state.

7.   (Currently Amended) The method as recited in claim 2, wherein transmitting the response data in accordance with the advisory state associated with the HTTP request when the advisory state is in the first state further comprises:

receiving second response data from the HTTP daemon; and

transmitting the second response data when the advise state is in a fourth state without transmitting the response data in the in-kernel HTTP cache and without storing the second response data in the in-kernel HTTP cache.

8.   (Withdrawn)

9.   (Withdrawn)

10. (Withdrawn)

11. (Withdrawn)

12. (Withdrawn)

13. (Withdrawn)

14. (Withdrawn)

15. (Withdrawn)

16. (Withdrawn)

17. (Withdrawn)

18. (Withdrawn)

19. (Withdrawn)

20. (Withdrawn)

21. (Withdrawn)

22. (Withdrawn)

23. (Withdrawn)

24. (Currently Amended)A computer-readable medium for managing a an in-kernel HTTP cache in a web server, the computer-readable medium storing computer-

readable instructions thereon, comprising:

instructions for receiving a HTTP request;

instructions for determining whether response data associated with the HTTP request is in the in-kernel HTTP cache;

instructions for when it is determined that response data associated with the HTTP request is in the in-kernel HTTP cache, obtaining an advisory state associated with the HTTP request from the in-kernel HTTP cache, the advisory state when in a first state indicating that it is necessary to obtain permission from a HTTP daemon to determine whether the response data can be transmitted and when in a second state indicating that the response data can be transmitted without obtaining permission from the HTTP daemon; and

instructions for transmitting the response data in accordance with the advisory state associated with the HTTP request.

Please **ADD** claims as follows:

25.   (Currently Amended) An apparatus for managing an in-kernel HTTP cache in a web server, comprising:

means for receiving a HTTP request;

means for determining whether response data associated with the HTTP request is in the in-kernel HTTP cache;

means for when it is determined that response data associated with the HTTP request is in the in-kernel HTTP cache, obtaining an advisory state associated with the HTTP request from the in-kernel HTTP cache, the advisory state when in a first state indicating that it is necessary to obtain permission from a HTTP daemon to determine whether the response data can be transmitted and when in a second state indicating that the response data can be transmitted without obtaining permission from the HTTP daemon; and

means for transmitting the response data in accordance with the advisory state associated with the HTTP request.

26.   (Currently Amended) An apparatus for managing an in-kernel HTTP cache in

a web server, comprising:

a processor; and

a memory, at least one of the processor and the memory being adapted for:

receiving a HTTP request;

determining whether response data associated with the HTTP request is in the in-kernel HTTP cache;

when it is determined that response data associated with the HTTP request is in the in-kernel HTTP cache, obtaining an advisory state associated with the HTTP request from the in-kernel HTTP cache, the advisory state when in a first state indicating that it is necessary to obtain permission from a HTTP daemon to determine whether the response data can be transmitted and when in a second state indicating that the response data can be transmitted without obtaining permission from the HTTP daemon; and

transmitting the response data in accordance with the advisory state associated with the HTTP request.